



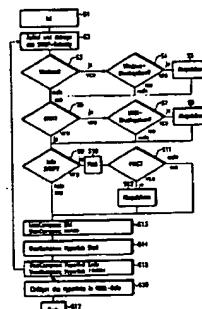
(51) Internationale Patentklassifikation ⁷ :	A2	(11) Internationale Veröffentlichungsnummer: WO 00/70837
H04L 29/00		(43) Internationales Veröffentlichungsdatum: 23. November 2000 (23.11.00)
(21) Internationales Aktenzeichen:	PCT/EP00/04312	(81) Bestimmungsstaaten: CA, JP, US, europäisches Patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).
(22) Internationales Anmeldedatum:	12. Mai 2000 (12.05.00)	
(30) Prioritätsdaten: 199 22 118.9	12. Mai 1999 (12.05.99)	DE
(71) Anmelder (für alle Bestimmungsstaaten ausser US): OCE PRINTING SYSTEMS GMBH [DE/DE]; Siemensallee 2, D-85586 Poing (DE).		Veröffentlicht <i>Ohne internationalen Recherchenbericht und erneut zu veröffentlichen nach Erhalt des Berichts.</i>
(72) Erfinder; und		
(75) Erfinder/Anmelder (nur für US): HOFSTETTER, Andreas [DE/DE]; Haselweg 5, D-82024 Taufkirchen (DE).		
(74) Anwälte: SCHAUMBURG, Karl-Heinz usw.; Postfach 86 07-48, D-81634 München (DE).		

(54) Title: NETWORK, INTERPRETER FOR SUCH A NETWORK, AND METHOD FOR OPERATING A NETWORK

(54) Bezeichnung: NETZWERK, INTERPRETER FÜR EIN DERARTIGES NETZWERK UND VERFAHREN ZUM BETREIBEN EINES NETZWERKES

(57) Abstract

The invention relates to a client-server network, to an interpreter that can be installed on the server of the network, and to a method for operating such a client-server network. The invention is characterized in that files are stored on the server which can be called up by the client and which, according to the invention, comprise both language elements that can be run on the client as well as language elements that can be run on the server. An interpreter is provided on the server which interprets the language elements that can be run on the server and which runs the same. According to a preferred embodiment of the invention, the language elements that can be run on the client correspond to a markup language, such as SGML, XML, HTML. When establishing these files, the user can then use known auxiliary means thereof to create the files, which are normally common word processing programs, in order to provide individual applications on the server of the network. Said applications can be called up by any client using a conventional browser. The invention is especially suited for controlling devices, in particular, printers and printing systems in addition to the corresponding pre-processing and post-processing devices due to the fact that the control intelligence is centrally stored on the server and, as a result, can be used by many clients. In addition, the data transfer between the clients and the server is held to a low level. The invention is additionally characterized in that a trading of, for example, print jobs between a plurality of servers can be implemented using simple means.

**(57) Zusammenfassung**

Die vorliegende Erfindung betrifft ein Client-Server-Netzwerk, einen am Server des Netzwerkes installierbaren Interpreter und ein Verfahren zum Betreiben eines derartigen Client-Server-Netzwerkes. Die Erfindung zeichnet sich dadurch aus, dass am Server Dateien abgelegt werden, die vom Client abgerufen werden können und die erfindungsgemäß sowohl am Client ausführbare Sprachelemente als auch am Server ausführbare Sprachelemente aufweisen. Am Server ist ein Interpreter vorgesehen, der die am Server ausführbaren Sprachelemente interpretiert und zur Ausführung bringt. Nach einer bevorzugten Ausführungsform der Erfindung entsprechen die am Client ausführbaren Sprachelemente einer Markup-Sprache, wie z. B. SGML, XML, HTML, da dann der Benutzer beim Einrichten dieser Dateien seine bekannten Hilfsmittel zum Erstellen der Dateien, die in der Regel geläufige Textverarbeitungsprogramme sind, verwenden kann, um am Server des Netzwerkes individuelle Anwendungen vorzusehen, die von einem beliebigen Client mit einem herkömmlichen Browser aufgerufen werden können. Die Erfindung eignet sich besonders zur Steuerung von Geräten, insbesondere von Druckern und Drucksystemen und den entsprechenden Vor- und Nachbearbeitungsgeräten, da die Steuerungsintrigenz zentral am Server hinterlegt wird und somit von vielen Clients benutzt werden kann, und der Datentransfer zwischen den Clients und dem Server gering gehalten wird. Ein weiterer Aspekt der Erfindung ist, dass mit einfachen Mitteln ein Trading von z.B. Druckaufträgen zwischen mehreren Servern realisiert werden kann.

LEDIGLICH ZUR INFORMATION

Codes zur Identifizierung von PCT-Vertragsstaaten auf den Kopfbögen der Schriften, die internationale Anmeldungen gemäss dem PCT veröffentlichen.

AL	Albanien	ES	Spanien	LS	Lesotho	SI	Slowenien
AM	Armenien	FI	Finnland	LT	Litauen	SK	Slowakei
AT	Österreich	FR	Frankreich	LU	Luxemburg	SN	Senegal
AU	Australien	GA	Gabun	LV	Lettland	SZ	Swasiland
AZ	Aserbaidschan	GB	Vereinigtes Königreich	MC	Monaco	TD	Tschad
BA	Bosnien-Herzegowina	GE	Georgien	MD	Republik Moldau	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagaskar	TJ	Tadschikistan
BE	Belgien	GN	Guinea	MK	Die ehemalige jugoslawische Republik Mazedonien	TM	Turkmenistan
BF	Burkina Faso	GR	Griechenland	ML	Mali	TR	Türkei
BG	Bulgarien	HU	Ungarn	MN	Mongolei	TT	Trinidad und Tobago
BJ	Benin	IE	Irland	MR	Mauretanien	UA	Ukraine
BR	Brasilien	IL	Israel	MW	Malawi	UG	Uganda
BY	Belarus	IS	Island	MX	Mexiko	US	Vereinigte Staaten von Amerika
CA	Kanada	IT	Italien	NE	Niger	UZ	Usbekistan
CF	Zentralafrikanische Republik	JP	Japan	NL	Niederlande	VN	Vietnam
CG	Kongo	KE	Kenia	NO	Norwegen	YU	Jugoslawien
CH	Schweiz	KG	Kirgisistan	NZ	Neuseeland	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Demokratische Volksrepublik Korea	PL	Polen		
CM	Kamerun	KR	Republik Korea	PT	Portugal		
CN	China	KZ	Kasachstan	RO	Rumänien		
CU	Kuba	LC	St. Lucia	RU	Russische Föderation		
CZ	Tschechische Republik	LI	Liechtenstein	SD	Sudan		
DE	Deutschland	LK	Sri Lanka	SE	Schweden		
DK	Dänemark	LR	Liberia	SG	Singapur		

Netzwerk, Interpreter für ein derartiges Netzwerk und Verfahren
5 zum Betreiben eines Netzwerkes

Die Erfindung betrifft ein Netzwerk, einen Interpreter für ein
derartiges Netzwerk und ein Verfahren zum Betreiben eines Netz-
10 werkes. Insbesondere betrifft die Erfindung ein Netzwerk für
ein Drucksystem.

An den bestehenden Datennetzen, wie z.B. LANs (Local Area Net-
works) und WANs (Wide Area Networks) sind in der Regel mehrere
15 Drucker und Drucksysteme und eventuell entsprechende Vor- und
Nachbearbeitungsgeräte, die z.B. das bedruckte Papier auf ein
spezielles Format zurechtschneiden oder binden, angeschlossen.
Die einzelnen Drucker und Drucksysteme unterscheiden sich in
ihrer Leistungsfähigkeit stark, so können z.B. kleine Tinten-
20 strahldrucker mit einer Druckleistung von 4 Seiten pro Minute
oder Hochleistungsdrucker mit einer Druckleistung von 40 Seiten
oder mehr pro Minute an das Datennetz angeschlossen sein. Die
Drucker und Drucksysteme unterscheiden sich nicht nur in ihrer
Druckleistung, sondern auch in der Druckqualität. So gibt es
25 z.B. Drucker die nur in einer einzigen Farbe (monochrom) druk-
ken, wohingegen auch zunehmend Farbdrucker zum Einsatz kommen.
Elektrophotographische Hochleistungsdruckgeräte sind heutzutage
ohne weiteres in der Lage, in zwei Farben zu drucken, das
heißt, in einem sogenannten Spot-Color-Betrieb oder Highlight-
30 Color-Betrieb. Ein derartiger Drucker ist beispielsweise von
der Océ Printing Systems GmbH unter der Bezeichnung PAGESTREAM®
200DSC bekannt.

Die unterschiedlichen Drucker und Drucksysteme sind in der Re-
35 gel auch mit unterschiedlichen Systemschnittstellen an das je-
weilige Netzwerk gekoppelt, wie z.B. dem SNMP (Simple Network
Management Protokoll) oder dem DMI (Desktop Management Inter-

face). Obwohl oftmals über das Netzwerk zu mehreren Druckern bzw. Drucksystemen eine physikalische Verbindung besteht, kann nur ein Teil der Drucker angesprochen werden.

5 Mit der rasanten Entwicklung der Intra- und Internets, mit welchen mehrere LANs und WANs zu einem einzigen Netz verbunden sind, insbesondere durch die Einführung des auf dem HTTP-Protokoll (**Hypertext Transport Protokoll**) basierenden WWW-Dienstes, steigt die Anzahl der Drucker und Drucksysteme, die 10 von einem einzigen Benutzer grundsätzlich angesprochen werden können explosionsartig an, wobei auch die Vielfalt der Protokolle zum Ansprechen der Drucker und Drucksysteme entsprechend zunimmt.

15 Dieser Erfolg des Internet und der Intranets wurde stark durch die Einführung des World Wide Web (WWW) gefördert, das von Internet-Benutzern einfach bedient werden kann und erlaubt, Informationen unterschiedlichster Art und von unterschiedlichsten Orten der gesamten Welt abzurufen. Der WWW-Dienst des Internets 20 beruht auf dem Client-Server-Prinzip. Die Kommunikation erfolgt zwischen einem Web-Server, der auch als WWW-Server bezeichnet wird und der Informationen bereitstellt, und einem Client, der die Informationen anzeigt. Die Informationen sind auf dem Web-Server in Seiten gespeichert, wobei die Daten in den Seiten z. 25 B. im sogenannten HTML-Format (**Hypertext Markup Language**) abgespeichert sind. Es sind auch andere Formate gebräuchlich, wie z. B. XML. Diese Formate sind Derivate des grundlegenden SGML-Formates (**Standard General Markup Language**). Diese Formate werden als Markup-Sprachen bezeichnet, da mit ihnen die Aufmachung 30 beschrieben und festgelegt werden kann.

Ein entsprechendes Dokument besteht aus normalem Text, bei dem Steueranweisungen, sogenannte „Tags“ in den Text eingefügt sind. Diese Tags beeinflussen unter anderem das Layout, das 35 später im Betrachtungsprogramm, dem Browser, am Client angezeigt wird. So gibt es z.B. Tags um Überschriften zu erzeugen,

WO 00/70837

oder Tags, die das Schriftbild verändern können. Die Tags werden immer in „< ... >“ eingeschlossen.

Die Übertragung der Informationen vom Web-Server zum Client er-
5 folgt gemäß dem HTTP-Protokoll, wobei die an den Client über-
tragene Information vom dort installierten Browser gelesen und
die einzelnen Tags interpretiert werden, so daß die Information
in der vorbestimmten Art und Weise am Bildschirm des Client
dargestellt werden.

10 Da mit dem HTML-Format im wesentlichen nur statische Bilder und
Texte dargestellt werden können, ist der WWW-Dienst zur Einbin-
dung von Multimedia-Elementen, Animationen oder Programmen
durch JAVA ergänzt worden. JAVA ist eine von SUN Microsystems
15 entwickelte, objektorientierte Programmiersprache für Web-
Anwendungen. JAVA unterstützt Text-, Hypertext-, Grafik-, Au-
dio- und Animationsfunktionen.

Die in JAVA programmierten Programmpakete können entweder als
20 sogenannte JAVA Applets vom Web-Server auf den Client geladen
werden oder am Client als sogenannte JAVA-Applikationen vorin-
stalliert sein. Die JAVA-Applets sind am Server separat von den
dazugehörigen HTML-Seiten gespeichert. Eine Abwandlung von JA-
VA, die als JAVAscript bezeichnet wird, erlaubt das Einfügen
25 von Programmpaketen in der HTML-Seite.

Um über das Internet Drucker und Drucksysteme ansteuern zu kön-
nen, hat man JAVA-Applikationen entwickelt, mit welchen man so-
wohl Verwaltungsfunktionen an den Druckern und Drucksystemen
30 als auch das Absenden von Druckaufträgen an die einzelnen Druk-
ker bzw. Drucksysteme ausführen kann. Diese JAVA-Programme
stellen einen wesentlichen Fortschritt gegenüber dem bisherigen
Zustand dar, da es dem Benutzer eines Client-Rechners ermög-
licht, über das Internet mehrere Drucker und Drucksysteme
35 plattformunabhängig anzusteuern. Die Art der Drucksysteme, das
heißt, die Schnittstellentypen, die von einem Client angesteu-
ert werden können, sind durch die jeweilige JAVA-Applikation

festgelegt, wobei für jeden Schnittstellentyp ein separates Programmelement vorgesehen ist. Dies bedeutet, daß je mehr unterschiedliche Schnittstellen mit einer solchen JAVA-Applikation angesteuert werden sollen, desto umfangreicher muß diese JAVA-Applikation sein. Entsprechendes gilt für den Funktionsumfang, der bei einer solchen JAVA-Applikation angesteuert werden soll. Dies hat zur Folge, daß diese JAVA-Applikationen umfangreiche Programme mit z.B. einen Datenumfang von 9 MB sind. Sie beanspruchen somit am Client erhebliche Rechenleistung und Speicherplatz und sind deshalb nur für Anwender zweckmäßig, die regelmäßig unterschiedliche Drucker und Drucksysteme ansteuern wollen.

Es ist auch bekannt, am Server ausführbare Programme vorzusehen, die vom Client aufgerufen werden können. Diese Programme sind entweder als vollständig compilierte Programme oder als von einem Interpreter interpretierbarer Programmcode am Server hinterlegt. Für einen derartigen interpretierbaren Programmcode wird für Internetanwendungen häufig die Programmiersprache Perl verwendet. Sie ist insbesondere für am Server auszuführende Datenbankanwendungen geeignet. Mit Perl können jedoch auch vom Server zum Client ladbare Seiten erzeugt werden, wobei Perl Befehle vorsieht, mit welchen die Aufmachung entsprechender Seiten aufgebaut werden kann. Hierbei kann auch vom Perl-Programmcode ein HTML-Programmcode generiert werden. Beim Aufruf eines derartigen Programmes am Server wird der vollständige Programmcode am Server interpretiert und ausgeführt. Auf Perl basierende Programme zum Ansteuern von Druckern, Drucksystemen und entsprechenden Vor- und Nachbearbeitungsgeräten sind dem Erfinder der vorliegenden Erfindung nicht bekannt.

In den Veröffentlichungen Jörn Heid, „Kettenreaktion“ in ix 11/1998, Seiten 166-171, von Jörn Heid, „Was es sein darf“, in ix 11/1998, Seiten 64-67 oder von Rainer Klute, „Mehr als Applets“ in: ix 11/1998, Seiten 60-63 werden Servlets beschrieben, die auf einem Server abgespeicherte Java-Programme sind. Die Servlets können in HTML-Seiten eingebunden werden und werden

beim Aufruf dieser Seiten, am Server ausgeführt. Mit derartigen Servlets können z.B. Teile oder vollständige HTML-Seiten erzeugt werden. Servlets stellen somit eine Möglichkeit dar, am Server ausführbare Java-Programme in HTML-Seiten einzubinden.

5

Aus der Veröffentlichung von Jörn Heid, „Hand angelegt“ In: ix 11/1998, Seiten 68-70 gehen sogenannte Java Server Pages (JSP) hervor, bei welchen Script- und HTML-Code im selben Dokument enthalten sein kann, wobei vor dem Verschicken an den Client die Scripts durch deren Ergebnis ersetzt werden. Mit dieser Technik ist es somit möglich, am Server Dateien vorzusehen, die vom Client aufgerufen werden und sowohl am Client als auch am Server ausführbare Sprachelemente umfassen. Ähnliche Techniken sind unter dem Markennamen Live Wire und Active Server Pages bekannt.

10

15

Die Veröffentlichung von B. Merkle und F. Pilhofer, „Perlen vor die Middleware“, in: ix 4/1999, Seiten 154-165 betrifft das CORBA-Mapping für Scriptsprachen. CORBA (Common Object Request Broker Architecture) ist eine aufwendige Architektur für einen Anfrage- und Vermittlungsdienst. Mit CORBA können alle möglichen Objekte innerhalb eines Netzwerkes vermittelt werden. Deshalb ist es auch möglich, auf Peripheriegeräten vorgesehene Objekte mittels CORBA zu vermitteln. Hierzu ist es jedoch notwendig, daß diese Objekte ein CORBA-spezifisches Protokoll zu Kommunikation beherrschen. Ferner weist CORBA ein API (Application Program Interface) auf, mit dem individuell gefertigte Programme am Server mit dem CORBA-System kommunizieren können. Diese individuellen Programme können selbstverständlich auch zur Kommunikation von Peripheriegeräten vorgesehen werden. Weiterhin ist es möglich, CORBA-unabhängige Protokolle auf das CORBA-Protokoll zu mappen. Hierdurch ist grundsätzlich auch eine Kommunikation zu anderen Protokollen möglich. Das Erstellen eines derartigen Mapping bzw. von eigenständigen Programmnen ist aufwendig. Das Erzeugen von CORBA-Objekten an Peripheriegeräten ist auch aufwendig, da das von CORBA vorgegebene Protokoll zur Kommunikation von Objekten eingehalten werden muß, das durch

seine allgemeine Anwendbarkeit sehr umfangreich und kompliziert ist.

Aus der DE 197 04 694 A1 ist ein Verfahren und eine Vorrichtung zum Steuern eines Peripheriegerätes über das Internet bekannt, das an sich bekannte CGI-Scripts verwendet, welche am Server unabhängig von einer HTML-Seite abgelegte Programme sind, die jedoch mit einer entsprechenden Referenz aus HTML-Seiten aufgerufen und am Server ausgeführt werden können.

10

Aus dem Druckerbuch der Océ Printing Systems GmbH, ISBN 3-00-001019-X, Ausgabe 3d, Oktober 1998 geht auf den Seiten 12-2 bis 12-8 ein Print-Server für professionelle, Internet-unabhängige Anwendungen hervor. Dieser Print-Server wird als PRISMApro® bezeichnet. Ein solcher Print-Server besteht aus einem leistungsfähigen Personalcomputer, mit einer entsprechenden Software, um einen oder mehrere Drucker, insbesondere einen Schnell- oder Hochleistungsdrucker ansteuern zu können. Ein solcher Print-Server ist in ein Netzwerk eingebunden und setzt die einkommenden Datenströme in entsprechende Druckdaten um. Je nach Ausführungsform kann der Print-Server Daten in den Förmaten AFDPS, Line/SF, PostScript, TIFF, PDF, LCDS, Line/O und PCL verarbeiten. Derartige Print-Servers werden insbesondere bei Datenbankanwendungen eingesetzt, wobei aus einer Datenbank eine große Datenmenge mit variablen Daten gedruckt wird.

Ferner ist im Druckerbuch der Océ Printing Systems GmbH, ISBN 3-00-001019-X Ausgabe 3d, Oktober 1998 auf den Seiten 14-2 bis 14-12 das Druck-Produktionskontrollsystem Océ Dómain® beschrieben. Bei diesem System werden ein oder mehrere Drucker mit Vor- und Nachverarbeitungsgeräten mittels eines Netzwerkes verbunden. Das System steuert und überwacht die Druckproduktion. Es ist eine netzwerkbasierende Client-Server-Lösung mit einer Datenbank, die auf einem zentralen Datenbankserver installiert ist. Alle im Produktionsprozeß erfaßten Maschinendaten und Betriebsdaten werden in dieser Datenbank gesammelt und für spätere Auswertungen den verschiedenen Clients bereitgestellt oder

können in kundenspezifische Anwendungen exportiert werden. Mit Océ Domain® werden Schnittstellen zu dem im Netzwerk ange- schlossenen Hochleistungsdruckern und den verschiedenen Hard- und Softwarekomponenten realisiert. Es werden die Industrie-
5 standards DMI (Desktop Management Interface), LMO (Large Mail- room Operations) und ODBC (Open Data Base Connectivity) unter- stützt. Océ Domain® ist ein Hochleistungssteuer- und Kontröll- system, das insbesondere bei Druckzentren zur Anwendung kommt, bei welchen Hochleistungsdrucker mit Geräten zur Vorverarbei-
10 tung und zur Nachverarbeitung gekoppelt sind.

Der Erfindung liegt die Aufgabe zugrunde, eine einfache techni- sche Lösung zur Ansteuerung unterschiedlicher Drucker und Drucksysteme zu finden.

15 Die Aufgabe wird durch ein Netzwerk mit den Merkmalen des Anspruchs 1, einen Interpreter für ein derartiges Netzwerk mit den Merkmalen des Anspruchs 10 und durch ein Verfahren zum Be- treiben eines Netzwerkes mit den Merkmalen des Anspruchs 16 ge-
20 löst. Vorteilhafte Ausgestaltungen der Erfindung sind in den Unteransprüchen angegeben.

Das Netzwerk gemäß der Erfindung ist ein Netzwerk zum Zusam- men- schluß von Rechnern, das zumindest einen Client mit zumindest 25 einem Server verbindet, wobei am Server gespeicherte Dateien vom Client durch Mitteilen einer Dateiadresse abgerufen werden können, wodurch eine entsprechende Datei an den Server übertra- gen wird. Diese vom Server an den Client übertragbaren Dateien enthalten Sprachelemente, die am Client ausgeführt werden. Am 30 Server ist ein Interpreter vorgesehen, der weitere, am Server ausführbare Sprachelemente, die in den am Server gespeicherten und vom Client abrufbaren Dateien enthalten sind, interpretie- ren und ausführen kann.
35 Mit der Erfindung werden somit am Server Dateien vorgesehen, die sowohl am Server ausführbare Sprachelemente als auch am Client ausführbare Sprachelemente enthalten. Hierdurch werden

wesentliche Vorteile erzielt, da es nicht mehr notwendig ist, wie es z. B. von am Server ausführbaren Perl-Programmen bekannt ist, am Client ausführbare Sprachelemente durch Sprachelemente einer anderen Sprache zu generieren, was mit erheblichem Pro-
5 grammieraufwand verbunden ist.

Erfindungsgemäß wird ein Gateway in einen Interpreter eines Servers integriert, wobei Sprachelemente zum Aufrufen des Gateways vorgesehen sind..

10

Hierdurch wird am Server eine Computersprache bereitgestellt, mit welcher unmittelbar Gateways aufgerufen werden können. Die Integration der Gateways in die Programmiersprache erlaubt die unmittelbare Ansteuerung aus der Programmiersprache von belie-
15 bigen Peripheriegeräten. Hierdurch kann ein Benutzer, der eine am Server abzuspeichernde Datei erstellt, mit dem Erstellen dieser Datei gleichzeitig die Steuerung der Peripheriegeräte behandeln. Dies ist bei herkömmlichen Systemen nicht möglich, da die entsprechenden Schnittstellen entweder nur durch von
20 Spezialisten erstellbaren Programmen als CGI-Script, Servlet oder dergleichen am Server hinterlegt werden können, oder an ein bekanntes Brokersystem aufwendig gemappt werden müssen.

Im Vergleich zu dem oben beschriebenen CORBA-System lässt sich die Erfindung wesentlich einfacher und mit wesentlich weniger Programmcode im Server realisieren, und bietet dem Benutzer we-
25 sentlich mehr Möglichkeiten in der Festlegung seiner Anwendun-
gen, da sie als Programmiersprache frei programmierbar ist.

30 Ein erfindungsgemäßes Netzwerk erlaubt somit jedem Nutzer, der über einfache Hilfsmittel zur Erstellung derartiger am Server gespeicherter Dateien verfügt, auch Peripheriegeräte, die un-
mittelbar oder über das Netz mit dem Server in Verbindung sind, anzusteuern. Insbesondere können einfach Drucker angesteuert
35 werden.

Entsprechen die am Client ausführbaren Sprachelemente einer Markup-language (wie z. B. SGML, HTML, XML) können die Dateien mit den meisten gebräuchlichen Editoren und Textverarbeitungsprogrammen editiert werden. Zur Erstellung einfacher Markup-

5 Dateien sind oftmals keine Kenntnisse über die Markup-Sprache notwendig, da die entsprechende Syntax automatisch vom Editor bzw. vom Textverarbeitungsprogramm eingefügt wird. Mit der Erfindung können somit komplexe Anwendungen mit einfach zu gebrauchenden Textverarbeitungsprogrammen erstellt werden. Er muß

10 lediglich in die ihm geläufigen vom Server aufrufbaren Dateien die am Server ausführbaren Sprachelemente hinzufügen. Die Kombination von am Client ausführbaren als auch am Server ausführbaren Sprachelementen in einer Datei stellt somit eine wesentliche Erleichterung für den Benutzer dar, der die Dateien am

15 Server einrichtet, da er mit seinen ihm vertrauten Mitteln diese Dateien erstellen kann und die am Client ausführbaren Befehle nicht in einer anderen Computersprache codieren muß, bei deren Ausführung sie künstlich erzeugt werden.

20 Durch das Vorsehen eines Interpreters, der die vom Client abgerufenen Dateien auf darin enthaltene Sprachelemente untersucht und gegebenenfalls diese Sprachelemente am Server interpretiert und ausführt, kann durch die Übertragung lediglich einer einzigen Adresse vom Client zum Server die Ausführung eines an sich

25 beliebigen Programmes am Server initiiert werden. Es wird somit eine Möglichkeit geschaffen, mit minimalem Programmieraufwand am Client, und minimalstem Datenvolumen, das zwischen dem Server und dem Client übertragen werden muß, beliebige am Server vorab in entsprechenden Dateien hinterlegte Programme, aufrufen

30 zu können.

Diese Programme können von mehreren Clients benutzt werden, so daß lediglich an einer einzigen Stelle im Netzwerk, am Server, ein entsprechendes Programm Paket hinterlegt werden muß.

35 Die mit der Erfindung erzielten Vorteile werden besonders deutlich, wenn die Datenübertragung im Netzwerk auf einer standar-

disierten Datenübertragung basiert, wie z.B. der gemäß dem HTTP-Protokoll, das bei den Intra- und Internets angewandt wird. Bei Verwendung eines derartigen Standards wird am Client lediglich ein herkömmlicher Browser ohne weitere zusätzliche Programme benötigt, um die entsprechenden Funktionen anzusteuern. Bisher wurden zum Ansteuern derartiger Funktionen, wie es eingangs erläutert worden ist, aufwendige Programme z. B. in JAVA auf den Clients geladen, wodurch lediglich an den Clients die gewünschten Funktionen angesteuert werden können, auf welche die entsprechenden Programme installiert oder geladen worden sind. Bei dem erfindungsgemäßen Netzwerk können hingegen von jedem Client, an dem ein Browser vorgesehen ist, die entsprechenden Programme am Server aufgerufen werden, ohne daß zusätzliche Programme auf den Client geladen werden müssen. Man benötigt lediglich eine Zugangsberechtigung zu dem jeweiligen Server.

Ein wesentlicher funktioneller Vorteil der Erfindung gegenüber den herkömmlichen Systemen liegt darin, daß es gemäß der Erfindung möglich ist, daß in den am Server abgespeicherten Dateien Programmelemente enthalten sind, die den Client automatisch zu einem weiteren Server vermitteln. Diese Vermittlung erfolgt z.B. einfach durch Übertragen der Adresse des weiteren Servers vom ursprünglichen Server zum Client, die dann am Client ausgeführt wird, das heißt zu weiteren Server geschickt wird, wodurch eine Verbindung zwischen dem weiteren Server und dem Client aufgebaut wird.

Diese Vermittlungsfunktion kann auch als automatische Vermittlungsfunktion ausgestaltet sein, wobei am Server eine Datei angelegt ist, die die Dienste anderer Server beinhaltet, so daß, wenn von einem Client ein bestimmter Dienst an diesem einen Server abgefragt wird, und dieser Server den Dienst nicht erfüllen kann, den Client automatisch zu einem weiteren Server vermittelt, der den Dienst erledigen kann. Für den Client stellt sich diese Funktion so dar, als ob sein Auftrag zwischen den Servern ausgehandelt wird, um denjenigen Server festzustel-

len, der ihn erfüllen kann. Man bezeichnet diese Funktion der automatischen Weitervermittlung deshalb auch als „Trading“..

Das Trading umfaßt auch die Möglichkeit, daß der Server als eine Art Durchgangs- bzw. Relaisstation fungiert, wobei er eine Verbindung zwischen dem Client und einer weiteren Arbeitsstation wie z.B. einem weiteren Server einer Druckstation oder der gleichen schaltet, bei der der Server zwischen der Arbeitsstation und dem Client geschaltet ist und die Daten entsprechend weiterleitet. Da der Server mit einem Gateway versehen ist, kann zwischen der weiteren Arbeitsstation und dem Server und dem Server und dem Client die Datenübertragung auf einem unterschiedlichen System bzw. unterschiedlichen Protokoll beruhen, wobei die Daten vom Server entsprechend übersetzt werden, damit eine reibungslose Datenübertragung möglich ist.

Ein Drucksystem mit einem solchen Client-Server-Netzwerk erlaubt bei der Hinterlegung entsprechender Dateien am Server jedem Client, der auf diesen Server zugreifen kann, die mittels des Servers ansteuerbaren Druckeinrichtungen und Vor-, und Nachbearbeitungseinrichtungen anzusprechen, ohne daß hierfür am Client eine spezielle Software installiert sein muß. Die einzelnen Druckaufträge können auch von einem Server an einen anderen Server weitervermittelt werden, um z.B. auf einem für diesen speziellen Druckauftrag besonders geeigneten Drucker ausgedruckt zu werden. Man kann den Server mit Gateways zum Umsetzen der Daten in spezielle Druckprotokolle oder auch andere Arten von Übertragungsprotokollen versehen, so daß mittels eines einzigen Servers auf unterschiedlichste Art und Weise an das Netzwerk angekoppelte Drucker angesteuert werden können.

Eine aus mindestens einem Drucker, mit evtl. einem Vor- und Nachbearbeitungsgerät bestehende Druckstation kann einfach an ein Netzwerk angekoppelt werden, indem die Geräte der Druckstation (Drucker, Vor- und Nachbearbeitungsgeräte) mit einem Server verbunden werden, der mit dem erfundungsgemäßen Interpreter versehen ist, und an dem entsprechende Programme zur Ansteue-

rung der einzelnen Geräte der Druckstation in Dateien abgespeichert sind, die vom Client abgerufen werden können. Die Geräte einer solchen Druckstation können dann über das Netzwerk von jedem beliebigen Client angesteuert werden, sofern eine entsprechende Zugangsberechtigung zum Server vorliegt. Die Erfindung schafft somit eine Möglichkeit, mit der man lediglich durch Hinzufügen eines Servers an eine bestehende Druckstation einen Netzzugang zur Druckstation erstellt, der mit äußerst einfachen technischen Mitteln angesprochen werden kann.

10

Das erfindungsgemäße Netzwerk ist zur Ansteuerung von Druckern und entsprechenden Vor- und Nachbearbeitungsgeräten ausgebildet. Es kann jedoch auch zur Ansteuerung beliebiger anderer Geräte verwendet werden, da das Grundprinzip der vorliegenden Erfindung, das Vorsehen eines Interpreters an einem Netz-Server, der vom Client aufgerufene Dateien interpretiert, auf weitere, beliebige Netzwerkanwendungen übertragen werden kann.

Nachfolgend wird die Erfindung beispielhaft anhand der beigefügten Zeichnungen näher erläutert. In denen zeigen:

Fig. 1 schematisch ein erfindungsgemäßes Netzwerk in einem Blockschaltbild,

Fig. 2 eine Schnittstellenabfrage in einem Flußdiagramm, das den Programmablauf des der als Anhang beigefügten Computerprogramms zeigt, und

Fig. 3 ein Tradingverfahren im Flußdiagramm.

30

Fig. 1 zeigt ein Ausführungsbeispiel eines erfindungsgemäßen Netzwerkes. Das Netzwerk weist einen ersten Web-Server 1 einen zweiten Web-Server 2 und einen ersten und zweiten Client 3, 4 auf. Der erste und zweite Web-Server und der erste Client sind mittels eines LANs über Datenleitungen 5, 6 miteinander verbunden, wobei die Datenleitung 5 den ersten Web-Server mit dem ersten Client und die Datenleitung 6 den ersten Web-Server 1 mit

dem zweiten Web-Server 2 verbindet. Auf dem LAN wird ein Intranet betrieben.

Zwischen dem ersten Web-Server 1 und dem zweiten Client 4 befindet sich eine feste physikalische Datenverbindung. Bei Bedarf wird z.B. über das Telefonnetz eine entsprechende Datenverbindung 7 aufgebaut wird, wobei das Internet als Übertragungsmedium verwendet wird.

Der erste Client 3 ist z.B. ein über das Intranet mit dem ersten Web-Server 1 verbundener Bürocomputer eines Anwenders des erfindungsgemäßen Netzwerkes, wohingegen der Client 2 ein zu Hause beim Anwender stehender Personalcomputer ist, mit dem sich der Anwender mittels eines Modems im Internet einwählen kann und eine Verbindung zum ersten Web-Server 1 herstellen kann. Da die Datenverbindung 7 nicht permanent vorliegt, ist sie in Fig. 1 gestrichelt eingezeichnet.

An beiden Clients 3, 4 ist jeweils ein Browser installiert, so daß die beiden Clients 3, 4 mit dem Web-Server 1 mit dem vom Internet bekannten Diensten kommunizieren können. Diese Dienste sind z.B. Telnet, das eine Terminal-Simulation ermöglicht, oder FTP, mit dem Dateien übertragen werden können. Bei dem vorliegenden Ausführungsbeispiel wird der Dienst des World Wide Web (WWW) verwendet, der bei Bedarf automatisch auf die weiteren Dienste, wie z.B. FTP, News, Telnet, Gopher, E-mail, usw. des Internets zurückgreift. An dem ersten Web-Server 1 sind zwei Geräte 8, 9 angeschlossen, wobei das Gerät 8 mittels einer seriellen Leitung (RS 232/V24) mit dem Web-Server 1 verbunden ist und zwischen dem Gerät 9 und dem Server 1 eine Verbindung 11 nach dem SNMP-Protokoll (Simple Network Management Protokoll) installiert ist. Das SNMP-Protokoll wird insbesondere zum Ansteuern von an einem Netzwerk angeschlossenen Geräten verwendet. Am Web-Server 1 ist ein Gateway 12 vorgesehen, das die über die SNMP-Verbindung 11 einkommenden Daten vom SNMP-Protokoll in das vom Web-Server 1 auf den Datenverbindungen 5 bis 7 angewandte Internet-Protokoll umsetzt bzw. in umgekehrter

Richtung die Daten gemäß dem Internet-Protokollen auf das SNMP-Protokoll umsetzt, wenn sie vom Web-Server 1 zum Steuergerät 9 übertragen werden. Ein Gateway im Sinne der Erfindung sind alle Hard- und Softwarekomponenten am Web-Server, die eine Verbindung zu einem sich vom Internet unterscheidenden Kommunikationssystem bzw. Netzwerkes herstellen. Am Web-Server 1 können mehrere weitere Gateways zu anderen Kommunikationssystemen vorgesehen sein, die z.B. auf dem DMI-Standard (Desktop Management Interface), dem LP-Standard (Line Printer); SLP (Service Location Protocol), dem IPP-Standard (Internet Printing Protokoll) beruhen. Es sind Gateways zum PJMweb möglich. PJMweb ist eine web-basierender Druckclient, der von Océ entwickelt worden ist. Ein Gateway kann auch eine Terminal-Emulation erzeugen. Insbesondere können Gateways vorgesehen werden, die zur Ansteuerung spezieller Druckerkommunikationssysteme ausgebildet sind.

Da über die Gateways eine Verbindung zu unterschiedlichen Netzwerken hergestellt werden kann, können an einen mit einem Gateway versehenen Server grundsätzlich beliebig viele Geräte angeschlossen bzw. vom Server angesteuert werden.

Der zweite Web-Server 2 ist mit einem weiteren Gerät 13 über eine Leitung 14 verbunden. Hierzu ist am zweiten Server 2 ein speziell für das Gerät 13 ausgebildetes Kommunikationsprogramm 15 installiert. Dieses Kommunikationsprogramm 15 ist ein kompliziertes Programm für ein sogenanntes Web-Based-Management. Derartige Programme sind jeweils für eine spezielle Anwendung ausgerichtet, wie z.B. der Ansteuerung des Gerätes 13, wodurch mit diesem Programm lediglich eine einzelne Verbindung zu dem Gerät 13 hergestellt und lediglich ein bestimmter Gerätetyp angesteuert werden kann.

Der erste und zweite Web-Server 1, 2 weisen neben anderen Dienstprogrammen ein WWW-Dienstprogramm 16 auf, das die aus dem Stand der Technik bekannten Funktionen aufweist, das heißt, daß es bei Empfang eines URL aus einem Datenspeicher 17 des Webervers 1, 2 die entsprechenden Seiten ausliest und an den Client

3, 4 schickt, der die URL abgeschickt hat. Erfindungsgemäß ist am Server 1, 2 ein Interpreter 18 vorgesehen, mit welchem die im Speicher 17 abgelegten Seiten vor dem Versenden zu einem Client 3, 4 interpretiert werden, das heißt, daß in den Seiten 5 enthaltene Sprachelemente vom Interpreter 18 ausgeführt werden. Die einzelnen Seiten sind im HTML-Format abgespeichert, das heißt, daß sie standardisierte Sprachelemente aufweisen, die vom Browser des Clients 3, 4 ausgeführt werden können. Die vom Interpreter 18 ausführbaren Sprachelemente sind unabhängig von 10 dem vom Browser ausführbaren Sprachelementen. Sie können auch als Ergänzung zum HTML-Format betrachtet werden, weshalb das Format der abgespeicherten Seiten auch als erweitertes HTML-Format bezeichnet werden kann.

15 Im Anhang ist ein Programmcode eines Programmbeispiels angegeben, das unten näher erläutert wird. Zunächst wird der Befehlsumfang des Interpreters kurz erläutert. Der Interpreter umfaßt, wie die meisten Computersprachen, Befehle zur Stringverarbeitung, Systembefehle und Strukturbefehle. Der Interpreter 20 kann je nach Bedarf mit weiteren Gruppen von Befehlen versehen werden. Zusätzlich zu den vorgenannten Gruppen von Befehlen weist der Interpreter einen Befehl auf, mit welchem neue Befehle des Interpreters erzeugt werden können. Dieser Befehl ist „addfunction“, mit dem ein beliebiger Befehl (Funktion) aus beliebigen auf dem Server zur Verfügung stehenden Befehlen (Maschinensprache, Betriebssysteme oder sonstige Hochsprache) aufgebaut werden kann. „addfunction“ wird nicht in den HTML-Seiten verwendet, sondern kann vom Benutzer in einer Programmentwicklungsumgebung aufgerufen werden.

30 Wichtige Stringverarbeitungsbefehle sind „userDefineString“, „userGetCGIString“, „userPreReplaceString“, „userPostReplaceString“ und „userCompose“. Mit „userDefineString“ können Stringvariablen definiert werden. Mit „userGetCGIString“ kann ein am Client eingegebener String eingelesen werden. Die Befehle „userPreReplaceString“ und „userPostReplaceString“ dienen zum Anordnen eines Strings an eine vorbestimmte Stelle der Sei-

te, wobei mit dem Befehl „userPreReplaceString“ der String unmittelbar nach Ausführung dieses Befehls an der entsprechenden Stelle angeordnet wird, wohingegen beim „userPostReplaceString“ der String erst nach Abarbeitung aller Sprachelemente der Seite an der vorbestimmten Stelle angeordnet wird. Der durch den Befehl „userPreReplaceString“ dargestellte String kann somit durch die weiteren vom Interpreter auszuführenden Sprachelemente noch verändert werden, wohingegen der durch den Befehl „userPostReplaceString“ angeordnete String am Ende plaziert wird, wodurch eine Veränderung des Strings nicht mehr möglich ist.

Einer der mächtigsten Systembefehle ist „userSystem (...)“, wobei in die Klammer ein Befehl des Betriebssystems des Servers oder eines weiteren auf dem Server installierten Programmes eingegeben werden kann. Hierdurch können mit dem Interpreter am Server eingerichtete Befehle und Programme aufgerufen werden. Dieser Befehl wird aus einer HTML-Seite aufgerufen.

20 Eine Untergruppe der Systembefehle sind Steuerungsbefehle, die z.B. zum Ansteuern eines bestimmten Druckers dienen. Diese Befehle entsprechen in der Regel dem jeweiligen Druckersystembefehlen, die als Befehl des Interpreters jeweils umgesetzt werden und durch weitere, höhere Steuerungsbefehle ergänzt werden 25 können.

Die Strukturbefehle dienen zum Erstellen einer Programmstruktur mit Verzweigungen, Schleifen und dergleichen. Entsprechende Strukturbefehle sind z.B. „userFor“, „userGoSub“ oder „userIf“.

30 Entsprechend der Syntax von HTML werden auch die vom Interpreter 18 ausführbaren Befehle in „<...>“ gesetzt.

Bei den Befehlen des Interpreters 18 unterscheidet man zwei 35 Klassen, nämlich eine mit Befehlen, die nicht direkt vom Client aufgerufen werden können, und eine weitere Klasse, deren Befehle direkt vom Client aufgerufen werden können. Von den oben

beispielhaft angegebenen Befehle können diejenigen nicht vom Client aufgerufen werden, die mit „user“ beginnen. Diese Unterteilung in zwei Klassen der Befehle dient der Sicherheit, denn wenn ein Anwender von einem beliebigen Client einen „userSystem“ Befehl auf dem Server erzeugen könnte, könnte er den Server nach Belieben manipulieren. Zur Vermeidung einer solchen Manipulation ist der Speicher 17 in einen freien Speicherbereich 17a und in einen gesperrten Speicherbereich 17b unterteilt. Im freien Speicherbereich können vom Client Seiten aufgerufen werden und diese ggf. mit Parametern versorgt werden, wohingegen der gesperrte Speicherbereich 17b Seiten enthält, die nicht direkt vom Client Parameter erhalten oder von ihm aufgerufen werden können. Diese Seiten werden lediglich vom Interpreter aufgerufen. Der Interpreter kontrolliert hierbei die Parameterübergabe und die Programmausführung. Der Interpreter 18 ist derart ausgebildet, daß sicherheitsrelevante Befehle nur ausgeführt werden, wenn sie im gesperrten Speicherbereich 17b abgespeichert sind. Ein Anwender an einem der Clients kann lediglich den Inhalt des freien Speicherbereiches 17a lesen und direkt adressieren. Die im gesperrten Speicherbereich 17b abgelegten Seiten können nur indirekt über den Interpreter adressiert werden.

Der Administrator des Servers kann im gesperrten Speicherbereich 17b des Servers die zum Ansteuern der Geräte 8, 9 und 13 notwendigen Steuerprogramme hinterlegen, die in der Regel sicherheitsrelevant sind.

Nachfolgend wird der im Anhang angegebene Programmcode näher erläutert, dessen Programmablauf in Fig. 2 in einem Flußdiagramm dargestellt ist. Im Schritt S1 „ini“ wird eine Initialisierung durchgeführt, wobei bestimmte Werte und Strings für jeweils abzufragende Server gesetzt bzw. definiert werden.

Mit dem Schritt S2 wird das SNMP-Gateway 12 aufgerufen, wobei durch den Wert „1“ angegeben wird, daß von diesem Gateway etwas eingelesen werden soll, wobei das Ergebnis in „SNMPVALUE“ ge-

speichert wird. Hierbei wird in Abhängigkeit von einer Benutzereingabe eine Adresse eines Datenbankeintrages (Managed Object einer MIB) abgefragt. Das Ergebnis dieser Abfrage stellt den Rechertyp dar, der über die SNMP-Schnittstelle angesprochen wird.

Anhand des in der Variablen „SNMPVALUE“ abgespeicherten Wertes wird im nächsten Schritt S3 geprüft, ob das Betriebssystem des über das SNMP-Gateway angesprochenen Rechners Windows ist.

10 Falls das Betriebssystem Windows ist, verzweigt der Programmablauf auf die Abfrage S4, bei der erneut ein SNMP-Wert eingelesen wird und anhand dieses Wertes geprüft wird, ob an dem Rechner ein vorbestimmtes, unter Windows lauffähiges Drucksystem („Imagestream“) eingerichtet ist. Falls die Abfrage S4 15 das Vorhandensein eines derartigen Drucksystems ergibt, wird dies im Schritt S5 abgespeichert.

Nach dem Speichervorgang im Schritt S5 bzw. falls eine der Abfragen S3 oder S4 verneint worden ist, geht der Programmablauf 20 auf den Schritt S6 über, mit dem geprüft wird, ob das Betriebssystem des über das SNMP-Gateway angesprochenen Rechners ein Unix-Betriebssystem ist. Ergibt die Abfrage, daß das Betriebssystem ein Unix-Betriebssystem ist, wird der Programmablauf auf den Schritt S7 verzweigt, mit dem wiederum ein SNMP-Wert eingelesen wird, anhand dessen geprüft wird, ob ein Unix- 25 Betriebssystem (PJM bzw. Prisma) an dem Rechner eingerichtet ist. Ergibt die Abfrage ein derartiges Drucksystem, so wird dies im Schritt S8 abgespeichert.

30 Nach dem Speichervorgang des Schrittes S8 oder wenn eine der beiden Abfragen der Schritte S6 und S7 negativ ausgefallen ist, geht der Programmablauf auf einen Schritt S9 über, mit dem geprüft wird, ob der über das SNMP-Gateway angesprochene Rechner überhaupt ein mittels des SNMP-Protokolls ansteuerbarer Rechner 35 ist. Ergibt die Abfrage, daß der Rechner mittels des SNMP-Protokoll grundsätzlich ansteuerbar ist, so wird im Schritt S10 eine PING-Abfrage aufgerufen und im Schritt S11 das Ergebnis

der PING-Abfrage geprüft, ob der Rechner in Betrieb (on-line) ist. Ergibt diese Abfrage S11, daß der Rechner in Betrieb ist (on-line), so wird dies im Schritt S12 gespeichert, wohingegen, wenn der Rechner nicht in Betrieb (off-line) ist, dann entspricht dies der default-Einstellung, so daß keine zusätzliche Abspeicherung notwendig ist.

Die PING-Abfrage ist ein nicht nur im Internet vorhandener Programmabaustein, der an vielen Servern eingerichtet ist. Er wird deshalb mit der bereits oben beschriebenen Funktion „userSystem“ aufgerufen oder als eigenständiger Befehl in den Interpreter integriert. Eine erneute Programmierung dieses Programmabausteins erübrigt sich somit.

Nach dem Speichervorgang des Schrittes S12 oder falls die Abfrage im Schritt S9 ergab, daß der Rechner ein SNMP-Rechner ist oder falls die Abfrage im Schritt S11 ergab, daß der Rechner nicht in Betrieb ist, geht der Programmablauf auf den Schritt S13 über, mit dem ein das Drucksystem darstellendes Bild aufgenommen wird. Dies erfolgt mit dem Befehl „userCompose“, mit dem die durch die obigen Abfragen und Speichervorgänge belegte Variable „SNMPVALUE“ ausgewertet wird (siehe Zeile unterhalb des Befehles „userCompose“), wobei das Ergebnis „exclusiv“ ist. Dies ist in der darauffolgenden Zeile durch den Wert „false“ vorgegeben. Dies bedeutet, daß nur ein einziges Bild in die Variable „REPLACEMENT“ eingelesen werden kann. Je nach dem obigen Ergebnis, wird ein Bild für ein Imagestream-Drucksystem (imagestream.gif), ein Prisma-Drucksystem (Prisma.gif) oder ein Bild für eine erfolgreiche PING-Abfrage (ping.gif) oder ein Bild für den nicht-Betriebszustand des Rechners (offline.gif) oder ein Bild für einen mittels des SNMP-Protokoll ansprechbaren Rechners (SNMP.gif) eingelesen.

In den nachfolgenden Schritten S14 und S15 wird wiederum mittels des Befehls „userCompose“ eine Hyperlink-Start-Variable („SnmpHLStart“) bzw. eine Hyperlink-Ende-Variable („SnmpHLEnd“) mit den zum Erzeugen eines Hyperlinks notwendigen Strings be-

legt. Ein Hyperlink ist ein automatisierter Aufruf einer Datei des Servers, wobei ein Client die entsprechende URL an den die Datei aufweisenden Server sendet.

5 In Figur 2 ist gestrichelt ein Programmfpfeil vom Schritt S15 zum Schritt S2 eingezeichnet, der den Programmabschnitt zwischen den Schritten S2 und S15 zu einer Schleife schließt. Mit diesem Programmfpfeil wird ausgedrückt, daß dieser Programmabschnitt mehrfach zum Abfragen mehrerer über das SNMP-Gateway
10 erreichbare Rechner ausgeführt wird. Die Abfragen der einzelnen Rechner werden parallel ausgeführt. Dies ist möglich, da das SNMP-Gateway multitasking-fähig ist.

15 Im nachfolgenden Schritt S16 werden die den Hyperlink darstellenden Strings mittels des Befehls „userPostReplaceString“ in die Datei bzw. HTML-Seite an eine vorbestimmte Stelle eingefügt.

Mit dem oben beschriebenen Programm wird somit abgefragt, ob
20 mit dem SNMP-Gateway ein oder mehrere bestimmte Rechner ange- sprochen werden können und ob an einem der ansprechbaren Rechner ein bestimmtes Drucksystem eingerichtet ist. Ein entspre- chendes Bild wird dann eingelesen und ein auf den Rechner ver- weisender Link generiert und in der HTML-Seite abgelegt. Wird
25 nun die HTML-Seite zum Client übertragen, so kann entweder der Link manuell oder automatisch ausgeführt werden, womit eine Verbindung vom Client, der die Abfrage gestartet hat, direkt zum Rechner mit dem abgefragten Drucksystem hergestellt wird.

30 Der Client wurde somit von einem Server zu einem weiteren Ser- ver vermittelt, wobei der erste Server für den Client ein Drucksystem gesucht hat, mit dem vom Client aus dann direkt entsprechende Druckaufträge ausgeführt werden können. Die Such- und Vermittlungsprozedur ist vollständig am Server ausgeführt
35 worden, das zeigt, daß die notwendige „Intelligenz“ lediglich am Server vorgesehen sein muß und vom Client mit einem herkömm- lichen Browser abgefragt und bedient werden kann.

Das obige Programm ist lediglich ein stark vereinfachtes und verkürztes Beispiel für eine Abfrage von erreichbaren Drucksystemen und automatischer bzw. halbautomatischer Vermittlung an ein gewünschtes Drucksystem. Dieses Beispiel soll lediglich andeuten, welche Möglichkeiten durch das erfundungsgemäße Vorsehen eines Interpreters am Server geschaffen werden.

In Fig. 3 ist ein Flußdiagramm dargestellt, mit welchem gezeigt wird, wie ein Druckauftrag mit dem erfundungsgemäßen Netzwerk abgearbeitet werden kann. Ein Anwender gibt in einem Schritt S18 seinen Druckauftrag ein und ergänzt ihn durch bestimmte Wünsche an die Qualität der Druckerzeugnisses (farbig, Papierart, usw.) und notwendigen Druckmerkmalen (Anzahl der Kopien, Format, usw.).

Dieser Druckauftrag wird im Schritt S19 vom Client 3, 4 zum Server 1, 2 übertragen. Am Server wird eine Datenbasis, die die relevanten Daten für Verbindungen zu Druckern bzw. weiteren Servern mit angeschlossenen Druckern bzw. entsprechenden Vor- und Nachbearbeitungsgeräten enthält, aktualisiert S20. Die Aktualisierung S20 erfolgt, indem eines oder mehrere Gateways nach entsprechenden Drucksystemen abgefragt werden. Die hierbei ermittelten Daten werden in der im Server abgelegten lokalen Datenbasis eingetragen. Eine Umlaufzeit von ca. 10 Minuten ist vorgesehen. Diese aktualisierte Datenbasis wird nach den vom Anwender eingegebenen Parametern (Druckauftrag, Wünsche und Voraussetzungen) im Schritt S21 ausgewertet.

Anschließend wird mit einer Abfrage S22 geprüft, ob bei der Auswertung eine für den Druckauftrag geeignete Druckeinrichtung gefunden worden ist. Falls keine geeignete Druckeinrichtung gefunden worden ist, zweigt der Programmablauf auf den Schritt S23 ab, mit dem an den Client eine Nachricht gesandt wird, daß der Druckauftrag nicht ausführbar ist.

Ergibt die Abfrage im Schritt S22 hingegen, daß eine geeignete Druckeinrichtung vorhanden ist, wird mit einer weiteren Abfrage S24 geprüft, ob der Druckauftrag vom Server ausgeführt werden kann. Falls das Ergebnis dieser Abfrage nein ist, wird, wie es
5 anhand des Programmbeispiels aus Fig. 2 dargestellt worden ist, der Client zu einem weiteren Server vermittelt S25, der den Druckauftrag ausführen kann. Hierbei wird vorzugsweise der Druckauftrag zusammen mit den Parametern, (Wünsche und Voraus-
setzungen) direkt an den weiteren Server übermittelt, wobei die
10 Parameter bei Bedarf entsprechend modifiziert werden können.

Ergibt die Abfrage S24 hingegen, daß der Druckauftrag vom vorliegenden Server ausgeführt werden kann, wird in einer weiteren Abfrage S26 geprüft, ob der Druckauftrag direkt an die Druckeinrichtung vermittelt werden kann. Besitzt die Druckeinrichtung eine entsprechende Netzankopplung, so kann ein direkter Link auf die Druckeinrichtung erzeugt werden, wodurch der Server nicht weiter durch den Druckauftrag belastet ist.
15

20 Ergibt die Abfrage aus dem Schritt S26, daß eine direkte Vermittlung des Druckauftrages an die Druckeinrichtung möglich ist, so wird diese im Schritt S27 ausgeführt.

Ist eine solche Vermittlung nicht möglich, so wird mit dem
25 Schritt S28 ein Link auf den Server bzw. sein Gateway gesetzt, um die Druckdaten im Schritt S29 über das Gateway zur Druckeinrichtung zu übermitteln. Nach Beendigung des Druckvorganges erfolgt eine Druckbestätigung S30 vom Server zum Client, womit das Verfahren beendet ist.
30

Bei diesem Verfahren wird der Druckauftrag automatisch zu einer geeigneten Druckeinrichtung weitergeleitet. Es wird sozusagen zwischen einzelnen Servern und Druckeinrichtungen ausgehandelt, wer zur Abarbeitung des Druckauftrages zur Verfügung steht und
35 geeignet ist. Man bezeichnet deshalb ein solches Verfahren auch als „Trading“.

Wie es oben dargestellt ist, ist das erfindungsgemäße Netzwerk insbesondere zur Ausbildung eines dezentralisierten Drucksystems geeignet, das einen oder mehrere Druckserver umfassen kann. Bei einer Internet-Anwendung des Druckservers kann jeder 5 Internet-Client, der eine entsprechende Zugangsberechtigung zum Server besitzt, diesen für seine Druckaufträge benutzen. Die technische Realisierung eines solchen dezentralisierten Drucksystems ist äußerst einfach und setzt lediglich die Installation eines erfindungsgemäßen Interpreters voraus, in dem die entsprechenden Befehle zum Ansteuern der Druckeinrichtungen bzw. 10 der Vor- und Nachbearbeitungsgeräte eingerichtet sind. In einem einzigen Netzwerk können mehrere Server mit einem erfindungsgemäßen Interpreter versehen sein. Hierdurch ist es auch möglich, daß ein Druckauftrag zwischen mehreren Servern weitergereicht 15 wird.

Die Erfindung ist jedoch nicht auf ein Drucksystem beschränkt, sondern kann zur Ansteuerung, Überwachung, Wartung, usw. von beliebigen Geräten verwendet werden. So bestehen zur Zeit erhebliche Bestrebungen Haushaltsgeräte netzwerkfähig zu machen. 20 Mit einem erfindungsgemäßen Server können sie von einem Anwender über das Internet von einer beliebigen Stelle überprüft, abgefragt und evtl. in Betrieb gesetzt werden. Grundsätzlich ist die Verwaltung und Steuerung aller technischen Geräte mit 25 dem erfindungsgemäßen Netzwerk möglich. Insbesondere eignet es sich zur Verwaltung von Datenverarbeitungssystemen, Telekommunikationssystemen und Vermittlungssystemen, wobei es insbesondere für überregionale Systeme von Vorteil ist, da der erfindungsgemäße Server von einer beliebigen Stelle des Netzwerkes. 30 aus angesteuert werden kann.

Ein weiterer, wesentlicher Vorteil des erfindungsgemäßen Netzwerkes ist, daß durch das Vorsehen eines Interpreters es nicht auf eine bestimmte Anwendung beschränkt ist, sondern durch die 35 Gestaltungsmöglichkeit einer Computersprache es möglich ist, mit hochspracheähnlichen Sprachelementen die jeweilige Anwendung zu generieren. Hierdurch besitzt das erfindungsgemäße Sy-

stem eine maximale Flexibilität. Für die meisten Anwendungen gibt es bereits spezialisierte Programmteile, die lediglich in den Interpreter integriert werden müssen. Derartige Programmteile sind z.B. Gateways, Druckertreiber, Spooler oder sonstige singuläre Steuerprogramme. Insbesondere können spezialisierte Kommunikationsprotokolle, wie SNMP oder DMI für die lokale Datenübertragung zu den Druckeinrichtungen ausgenutzt werden, da diese Protokolle im Gegensatz zu dem HTTP-Protokoll einen wesentlich geringeren Protokolloverhead, geringere Responsezeiten und eine hohe Leistung besitzen und eine einfache Anwendung erlauben. Diese spezialisierten Protokolle bzw. spezifischen Schnittstellen können einem Anwender zugänglich gemacht werden, der sich weder mit dieser Technik auseinandersetzen muß, noch auf seinem Client eine entsprechende, in der Regel sehr aufwendige Software installieren muß, um eine Kommunikation zu derartigen spezialisierten Systemen herstellen zu können.

Der erfindungsgemäße Interpreter kann auf einen Datenträger gespeichert sein und von diesem oder über ein Netzwerk in einen Server geladen werden.

Die Erfindung kann folgendermaßen kurz zusammengefaßt werden:

Die vorliegende Erfindung betrifft ein Client-Server-Netzwerk, einen am Server des Netzwerkes installierbaren Interpreter und ein Verfahren zum Betreiben eines derartigen Client-Server-Netzwerkes.

Die Erfindung zeichnet sich dadurch aus, daß am Server Dateien abgelegt werden, die vom Client abgerufen werden können und die erfindungsgemäß sowohl am Client ausführbare Sprachelemente als auch am Server ausführbare Sprachelemente aufweisen. Am Server ist ein Interpreter vorgesehen, der die am Server ausführbaren Sprachelemente interpretiert und zur Ausführung bringt.

35

Nach einer bevorzugten Ausführungsform der Erfindung entsprechen die am Client ausführbaren Sprachelemente einer Markup-

Sprache, wie z. B. SGML, XML, HTML, da dann der Benutzer beim Einrichten dieser Dateien seine bekannten Hilfsmittel zum Erstellen der Dateien, die in der Regel geläufige Textverarbeitungsprogramme sind, verwenden kann, um am Server des Netzwerks individuelle Anwendungen vorzusehen, die von einem beliebigen Client mit einem herkömmlichen Browser aufgerufen werden können.

Die Erfindung eignet sich besonders zur Steuerung von Geräten, insbesondere von Druckern und Drucksystemen und den entsprechenden Vor- und Nachbearbeitungsgeräten, da die Steuerungsin-
telligenz zentral am Server hinterlegt wird und somit von vielen Clients benutzt werden kann, und der Datentransfer zwischen den Clients und dem Server gering gehalten wird.

Ein weiterer Aspekt der Erfindung ist, daß mit einfachen Mitteln ein Trading von z. B. Druckaufträgen zwischen mehreren Servern realisiert werden kann.

Anhang

```

<head>
<TITLE>SNMP Response</TITLE>
5  </HEAD>

<BODY>

<!--
10   Kommentar: Programmanfang
-->

15  <!--
-->
16  Kommentar: Funktion "STDefineString": wird von ini1 aufgerufen.
-->
20
25  <PSUB FUNC="STDefineString">
    <PTAG FUNC="userDefineString" VALUE="SNMPVALUE%i%" VALUE="Offline"></PTAG>
    <PTAG FUNC="userDefineString" VALUE="REPLACEMENT%i%"></PTAG>
    <PTAG FUNC="userDefineString" VALUE="SnmpHLStart%i%"></PTAG>
    <PTAG FUNC="userDefineString" VALUE="SnmpHLEnd%i%"></PTAG>
    <PTAG FUNC="userDefineString" VALUE="Stdout%i%"></PTAG>
    <PTAG FUNC="userDefineString" VALUE="Stdin%i%"></PTAG>
    <PTAG FUNC="userDefineString" VALUE="Stderr%i%"></PTAG>
</PSUB>
30

<!--
35   Kommentar: Funktion "MTScanSystem"
-->

40  <PSUB FUNC="MTScanSystem"> // Kommentar: Abschnitt s2
    <PTAG FUNC="userSNMPGateway"
        VALUE="SNMPVALUE%i%" VALUE="1"
        VALUE="160.120.17.%i%" VALUE="public"
        VALUE=".1.3.6.1.2.1.1.1.0">
45  </PTAG>

    <PBRANCH FUNC="userIf" // Kommentar: Abschnitt s3
        VALUE="SNMPVALUE%i%" VALUE="FNC>WIN"
        VALUE=<PTAG FUNC="userSNMPGateway" // Kommentar: Abschnitt s4
            VALUE="SNMPVALUE%i%" VALUE="1"
            VALUE="160.120.17.%i%" VALUE="public"
            VALUE=".1.3.6.1.4.1.1552.92.2.0">
55  </PTAG>
    </PBRANCH> // Kommentar: Abschnitt s5

60  <PBRANCH FUNC="userIf" // Kommentar: Abschnitt s6
        VALUE="SNMPVALUE%i%" VALUE="FNC>SCO"
        VALUE=<PTAG FUNC="userSNMPGateway" // Kommentar: Abschnitt s7
            VALUE="SNMPVALUE%i%" // Kommentar: Abschnitt s8
            VALUE="1"
65

```

```
        VALUE="160.120.17.%i%"  
        VALUE="public"  
        VALUE=".1.3.6.1.3.1.1.1.1.1.0">  
    </PTAG>  
5     </PBRANCH>  
  
<PBRANCH FUNC="userIf"                                // Kommentar: Abschnitt s9  
      VALUE="SNMPVALUE%i%"  
      VALUE="FNC>Offline"  
10    VALUE=<PTAG FUNC="userSystem"                  // Kommentar: Abschnitt s10  
          VALUE="Stdout%i%"  
          VALUE="Stdin%i%"  
          VALUE="Stderr%i%"  
          VALUE="0"  
15    VALUE="5000"  
          VALUE="ping -n 1"  
          VALUE="160.120.17.%i%">  
    </PTAG>  
20    </PBRANCH>  
  
<PBRANCH FUNC="userIf"                                // Kommentar: Abschnitt s11  
      VALUE="Stdout%i%"  
      VALUE="FNC>Antwort"                         // Kommentar: Abschnitt s12  
25    VALUE=<PTAG FUNC="userDefineString"           // Kommentar: Abschnitt s12  
          VALUE="SNMPVALUE%i%"  
          VALUE="PING">  
    </PTAG>  
20    </PBRANCH>  
  
30    <PTAG FUNC="userCompose"                      // Kommentar: Abschnitt s13  
      VALUE="SNMPVALUE%i%"  
      VALUE="false"  
  
      VALUE="REPLACEMENT%i%"  
35    VALUE="FNC>ISTREAM"  
      VALUE="..../images/imagesstream.gif"  
  
      VALUE="REPLACEMENT%i%"  
      VALUE="FNC>PRISMA"  
40    VALUE="..../images/prisma.gif"  
  
      VALUE="REPLACEMENT%i%"  
      VALUE="FCS>PING"  
45    VALUE="..../images/ping.gif"  
  
      VALUE="REPLACEMENT%i%"  
      VALUE="FNC>Offline"  
50    VALUE="..../images/offline.gif">  
    </PTAG>                                         // Kommentar: Abschnitt s14  
  
<PTAG FUNC="userCompose"  
      VALUE="SNMPVALUE%i%"  
      VALUE="false"  
  
      VALUE="SnmpHLStart%i%"  
55    VALUE="FNC>ISTREAM"  
      VALUE="<a HREF=\"http://www.ops.de\">>"  
  
      VALUE="SnmpHLStart%i%"  
60    VALUE="FNC>PRISMA"  
      VALUE="<a HREF=\"http://160.120.17.%i%/pjm.html\> tar-  
      get=\"_blank\>"  
  
      VALUE="SnmpHLStart%i%"  
65    VALUE="FNC>Offline"  
      VALUE="">  
    </PTAG>
```

```
<PTAG FUNC="userCompose"                                // Kommentar: Abschnitt s15
  VALUE="SNMPVALUE%i%" 
  VALUE="false"
5   VALUE="SnmpHLEnd%i%" 
  VALUE="FNC>ISTREAM"
  VALUE="</a>" 

10  VALUE="SnmpHLEnd%i%" 
  VALUE="FNC>PRISMA"
  VALUE="</a>" 

15  VALUE="SnmpHLEnd%i%" 
  VALUE="FNC>Offline"
  VALUE="">
</PTAG>
</PSUB>

20 <!--
-----  
Kommentar: Funktion "STPrintSystem: wird von s16 aufgerufen
-----//-->
25 <PSUB FUNC="STPrintSystem">
  <PTAG FUNC="userPreReplaceString" VALUE="SnmpHLStart%i%"></PTAG>
  <img src=<PTAG FUNC="userPreReplaceString" VALUE="REPLACEMENT%i%"></PTAG>
width="80" height="80"
30   alt="160.120.17.%i%
<PTAG FUNC="userPreReplaceString" VALUE="SNMPVALUE%i%"></PTAG>">
  <PTAG FUNC="userPreReplaceString" VALUE="SnmpHLEnd%i%"></PTAG>
</PSUB>

35 <!--
-----  
Kommentar: ini1 - s1
-----//-->
40 <PTAG FUNC="userSetSNMPRetries" VALUE="2"></PTAG>
<PTAG FUNC="userSetSNMPTimeout" VALUE="2000"></PTAG>

<PTAG FUNC="userFor"
  VALUE="i"
  VALUE="200"
45  VALUE="255"
  VALUE="1"
  VALUE="0"
  VALUE="STDefineString">
</PTAG>
50

<!--
-----  
Kommentar: Schritte s2-s15 → Aufruf der entsprechenden
Unterprogramme der Abschnitten s2-s15
-----//-->
55 <PTAG FUNC="userFor"
  VALUE="i"
  VALUE="200"
  VALUE="255"
  VALUE="1"
  VALUE="-1"
  VALUE="MTScanSystem">
</PTAG>
```

```
<!--
-----
 Kommentar: Schritt s16
-----
5 //-->

<p>
<PTAG FUNC="userFor"
      VALUE="i"
10    VALUE="200"
      VALUE="255"
      VALUE="1"
      VALUE="0"
      VALUE="STPrintSystem">
15 </PTAG>
</p>

<!--
-----
20 Kommentar: Schritt s17 / Ende
-----
//-->

25 </BODY>
</HTML>
```

Bezugszeichenliste

5 1 erster Web-Server
2 zweiter Web-Server
3 erster Client
4 zweiter Client
5 Datenleitung (Intranet)
10 6 Datenleitung (Intranet)
7 Datenverbindung (Internet)
8 Gerät
9 Gerät
10 Serielle Leitung
15 11 SNMP-Verbindung
12 Gateway
13 Gerät
14 Leitung
15 Kommunikationsprogramm
20 16 WWW-Dienstprogramm
17 Speicher
17a freier Speicherbereich
17b gesperrter Speicherbereich
18 Interpreter
25

Verfahrensschritte

- S1 Initialisierung
- 5 S2 Aufruf und Abfrage des SNMP-Gateway
- S3 Windows?
- S4 Windows-Drucksystem?
- S5 Abspeichern
- S6 UNIX?
- 10 S7 UNIX-Drucksystem?
- S8 Abspeichern
- S9 kein SNMP?
- S10 PING
- S11 PING?
- 15 S12 Abspeichern
- S13 UserCompose: Bild
- S14 UserCompose: Hyperlink Start
- S15 UserCompose: Hyperlink Ende
- S16 Einfügen des Hyperlinks in HTML-Seite
- 20 S17 Ende
- S18 Eingabe des Druckauftrages
- S19 Übertragung vom Client zum Server
- S20 Aktualisierung einer Datenbasis
- S21 Auswertung der Datenbasis
- 25 S22 Geeignete Druckeinrichtung?
- S23 Nachricht an Client
- S24 Kann Server Druckauftrag ausführen?
- S25 Vermittlung an weiteren Server
- S26 Kann an Druckeinrichtung vermittelt werden?
- 30 S27 Vermittlung an Druckeinrichtung mit direktem Link
- S28 Link auf Server
- S29 Übermittlung über Gateway
- S30 Druckbestätigung

Ansprüche

5 1. Netzwerk zum Zusammenschluß von Rechnern, wobei zumindest
ein Rechner als Server (1, 2) und ein Rechner als Client
(3, 4) agiert, und am Server (1, 2) gespeicherte Dateien
vom Server (1, 2) zum Client (3, 4) übertragen werden,
wenn sie der Client (3, 4) durch Senden einer entsprechen-
10 den Dateiadresse an den Server (1, 2) abruft, und
die Dateien sowohl am Client (3, 4) ausführbare Sprachele-
mente als auch am Server (1, 2) ausführbare Sprachelemente
enthalten, und
am Server (1, 2) ein Interpreter zum Interpretieren und
15 Ausführen der am Server ausführbaren Sprachelemente vor-
handen ist, und
am Server (1, 2) ein Gateway (12) installiert ist, das ei-
ne Datenverbindung zu einem weiteren logischen und/oder
physikalischen System (Gerät, Netzwerk, Queue, Computer)
20 herstellen kann, wobei die Daten des weiteren Systems ein
anderes Format als die zwischen dem Server (1, 2) und dem
Client (3, 4) ausgetauschten Daten aufweisen, und das Ga-
teway (12) sowohl die einkommenden als auch die ausgehen-
den Daten in die entsprechenden Datenformate automatisch
25 umsetzt, und
wobei das Gateway (12) im Interpreter integriert ist und
durch Sprachelemente des Interpreters (18) aufrufbar ist.

2. Netzwerk nach Anspruch 1,
30 wobei mehrere Gateways (12) im Interpreter integriert sind
und durch Sprachelemente des Interpreters (18) aufrufbar
sind.

3. Netzwerk nach Anspruch 1 oder Anspruch 2,
35 wobei der Interpreter (18) am Server (1, 2) derart ausge-
bildet ist, daß die am Server ausführbaren Sprachelemente
nach dem Aufrufen der Dateien durch einen Client und vor

der Übermittlung der Dateien zum Client (3, 4) am Server (1, 2) ausgeführt werden.

4. Netzwerk nach einem der Ansprüche 1 bis 3,
5 wobei die Dateiadresse dem URL-Format entspricht und der Server (1, 2) ein Web-Server ist, so daß die Dateien mit einem am Client (3, 4) installierten Internet-Browser abgerufen werden können.
- 10 5. Netzwerk nach einem der Ansprüche 1 bis 4,
wobei die am Server (1, 2) gespeicherten und vom Client (3, 4) abrufbaren Dateien dem Format einer Markup-Sprache (SGML, XML, HTML) entsprechen, das durch die am Server ausführbaren Sprachelemente erweitert ist.
- 15 6. Netzwerk nach einem der Ansprüche 1 bis 5,
wobei am Server (1, 2) zumindest ein Gateway (12) installiert ist, das eine Datenverbindung zu einem weiteren logischen und/oder physikalischen System (Gerät, Netzwerk, Queue, Computer) herstellen kann, wobei die Daten des weiteren Systems ein anderes Format als die zwischen dem Server (1, 2) und dem Client (3, 4) ausgetauschten Daten aufweisen, und das Gateway (12) sowohl die einkommenden als auch die ausgehenden Daten in die entsprechenden Datenformate automatisch umsetzt.
- 20 7. Netzwerk nach Anspruch 6,
wobei jeweils ein Gateway (12) zum Umsetzen der Daten in eines oder mehrere der folgenden Formate vorgesehen ist:
25 SNMP, LP, PJMweb, ftp.
- 30 8. Netzwerk nach Anspruch 6 oder 7,
wobei das bzw. die Gateways (12) im Interpreter integriert sind und durch Sprachelementen des Interpreters (18) aufgerufen werden können.
- 35 9. Netzwerk nach einem der Ansprüche 1 bis 8,

wobei am Server (1, 2) Programme zum Ansteuern zumindest eines Druckers und/oder Vor- oder Nachverarbeitungsgeräte, wie z.B. eines Druckertreibers oder eines Spoolers installiert sind, und diese Programme vom Interpreter (18) aufgerufen werden können.

10. Interpreter für ein Netzwerk, insbesondere nach einem der vorhergehenden Ansprüche, der an einem Server (1, 2) eines Netzwerkes zum Zusammenschluß von Rechnern installierbar ist und zum Interpretieren und Ausführen von am Server (1, 2) ausführbaren Sprachelementen ausgebildet ist, die in am Server (1, 2) abgespeicherten Dateien enthalten sind, wobei diese Dateien von einem Client (3, 4) mittels der Übertragung einer Adresse abgerufen werden können und zusätzlich am Client (3, 4) ausführbare Sprachelemente enthalten.
11. Interpreter nach Anspruch 10, wobei der Interpreter einen Befehl (userCompose) zum Erzeugen von String-Einträgen in der Datei aufweist.
12. Interpreter nach Anspruch 10 oder 11, mit einem Befehl (userPreReplaceString, userPostReplaceString) zum Setzen von String-Einträgen an eine vorbestimmte Stelle der Datei.
13. Interpreter nach einem der Ansprüche 10 bis 12, mit einem Befehl zum Einlesen eines vom Client (3, 4) an den Server (1, 2) übertragenen Strings und zum Abspeichern des Strings in eine vorbestimmte Variable (userGetCGIString).
14. Interpreter nach einem der Ansprüche 10 bis 13, mit einem Befehl zum Aufrufen eines Gateways und Abfragen eines mit dem Gateway verbundenen Systems.
15. Interpreter nach einem der Ansprüche 10 bis 14,

wobei der Interpreter eine Gruppe von Client-Befehlen umfaßt, die sowohl vom Client aus als auch vom Server aus aufgerufen werden können, und eine Gruppe von Serverbefehlen umfaßt, die nur vom Server aus aufgerufen werden können.

5

16. Interpreter nach einem der Ansprüche 10 bis 15, wobei der Interpreter auf einen Datenträger gespeichert ist.
17. Verfahren zum Betreiben eines Netzwerkes zum Zusammenschluß von Rechnern nach einem der Ansprüche 1 bis 9, wobei an einem Server (1, 2) gespeicherte Dateien vom Server (1, 2) zum Client (3, 4) übertragen werden, wenn sie der Client (3, 4) durch Senden einer entsprechenden Dateiadresse an den Server (1, 2) abruft, und die Server den Clients jeweils einen oder mehrere Dienste anbietet, wobei bei einer Client-Anfrage nach einem bestimmten Dienst mit bestimmten, den Dienst zugrundeliegenden Parametern, der abgefragte Server bestimmt, ob er den Dienst erfüllen kann, und wenn der Server feststellt, daß er den Dienst nicht erfüllen kann, er einen weiteren Server oder an ein am Netzwerk angeschlossenes Gerät an den Client vermittelt, das den Dienst ausführen kann.
18. Verfahren zum Betreiben eines Netzwerkes nach Anspruch 17, wobei einer der von den Servern angebotenen Dienste die Ausführung eines Druckauftrages ist, und der Server den Druckauftrag an einen anderen Server oder direkt an eine Druckeinrichtung weiterleitet, wenn der Server den Druckauftrag nicht selbst ausführen kann.
19. Verfahren zum Betreiben eines Netzwerkes nach Anspruch 17 oder 18, wobei der Server eine Datenbank aufweist, in der Informationen zu den im Netzwerk angebotenen Diensten abgespeichert sind.

chert sind, so daß bei einer Client-Anfrage anhand dieser Datenbanken ermittelt werden kann, ob der gewünschte Dienst im Netzwerk vorhanden ist.

5 20. Verfahren zum Betreiben eines Netzwerkes nach einem der Ansprüche 17 bis 19,
wobei die Vermittlung an einen weiteren Server oder ein an das Netzwerk angeschlossenes Gerät durch Erzeugen der Adresse des weiteren Servers oder des Gerätes und durch
10 Übermittlung der Adresse an den anfragenden Client ausgeführt wird;

21. Verfahren zum Betreiben eines Netzwerkes nach einem der Ansprüche 17 bis 20,
15 mit einem Interpreter nach einem der Ansprüche 10 bis 16.

1/3

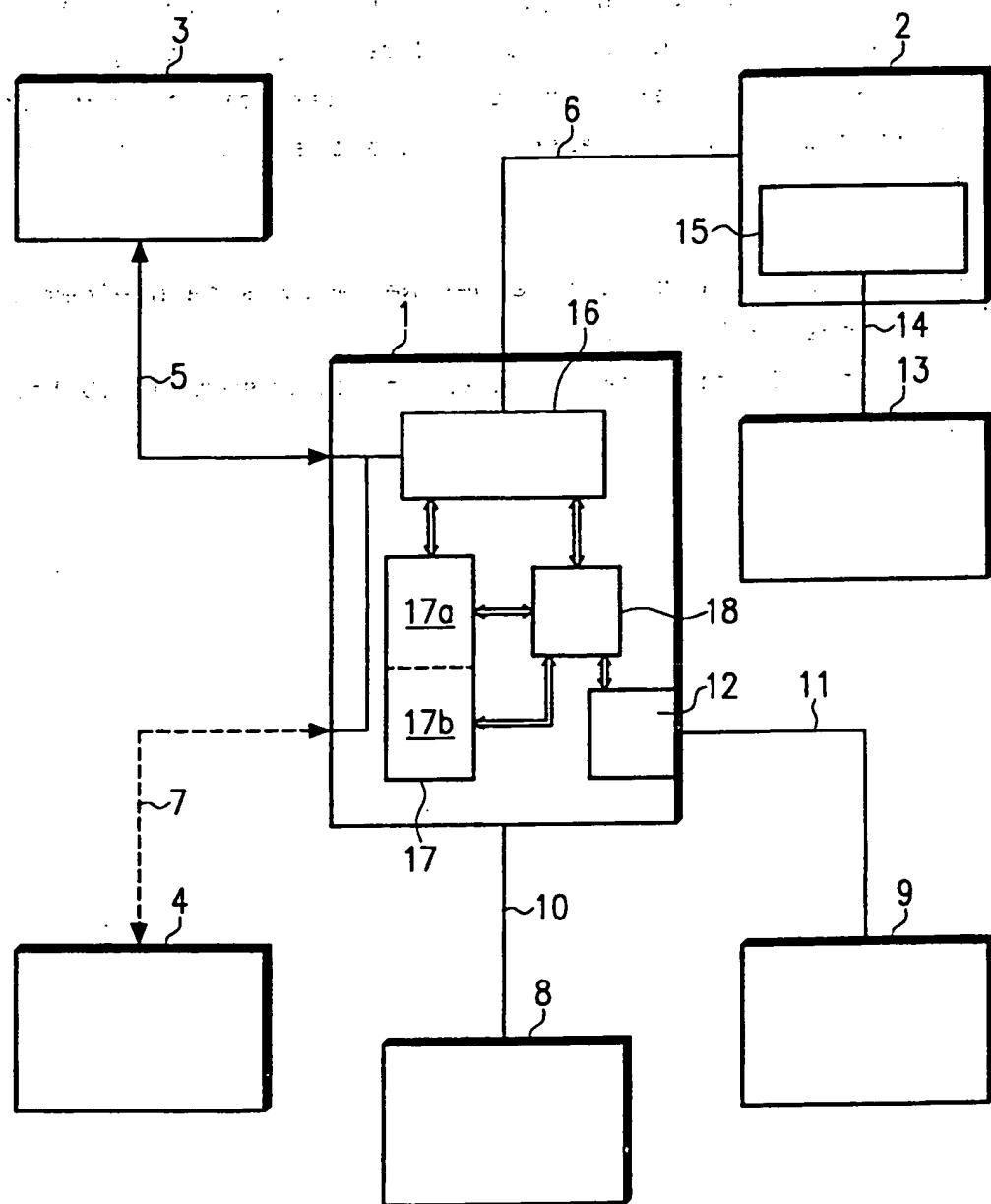


FIG. 1

2/3

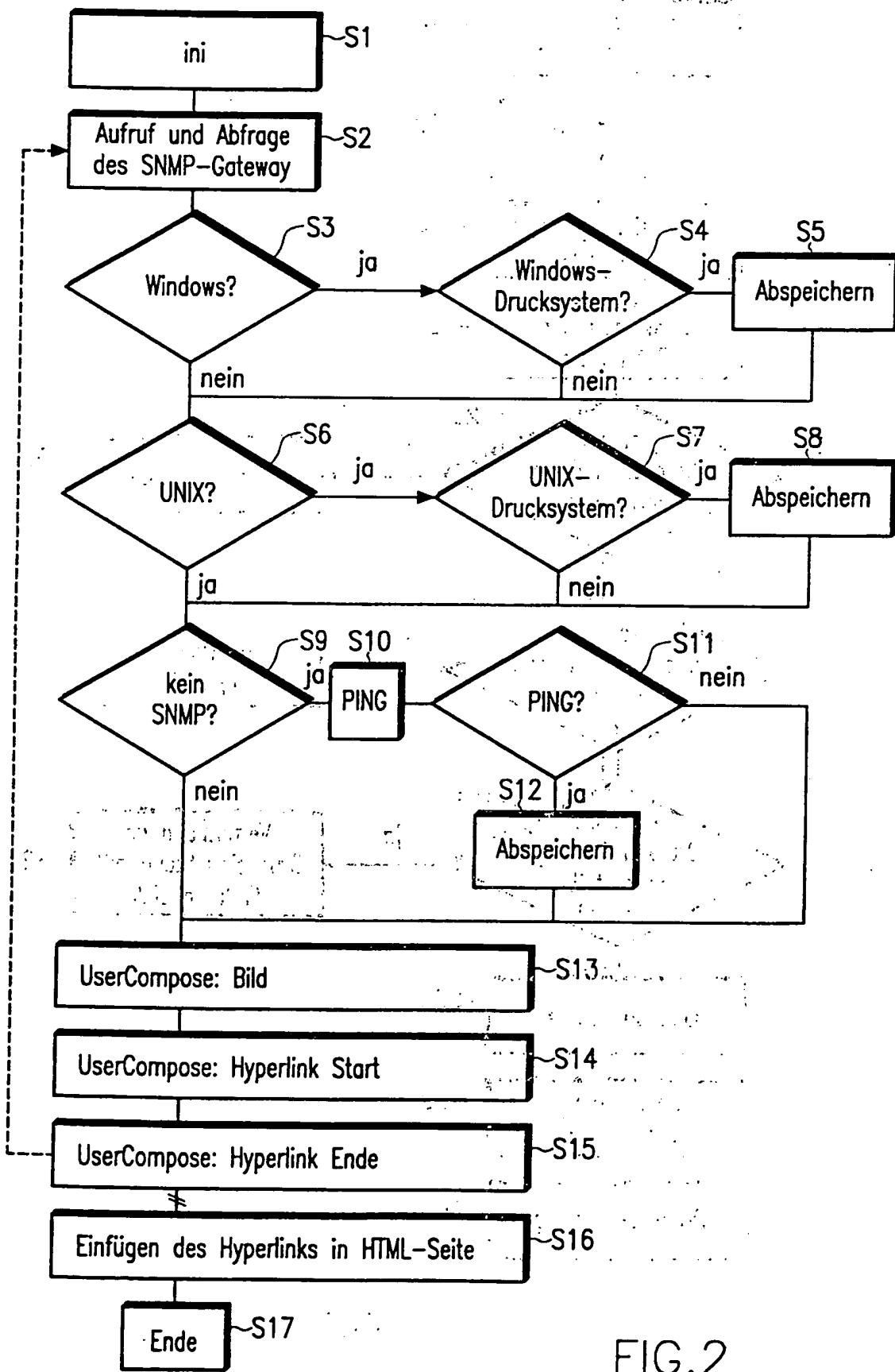


FIG.2

3/3

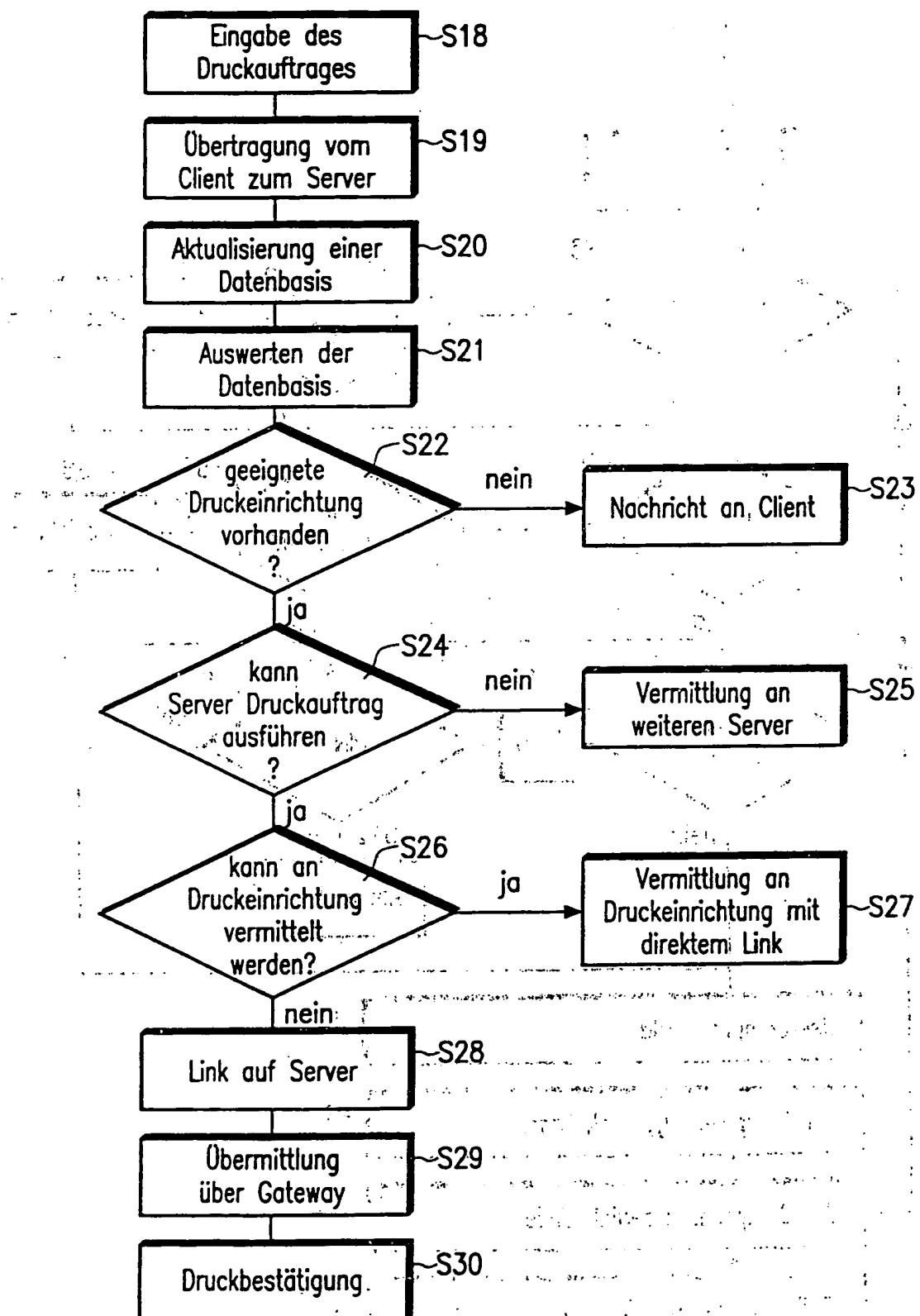


FIG.3

THIS PAGE BLANK (USPTO)